HIGH SPEED CONFIGURABLE TRANSCEIVER ARCHITECTURE

FIELD OF THE INVENTION

[0001] The present invention relates to data communication, and more specifically to a configurable high speed transceiver.

BACKGROUND OF THE INVENTION

[0002] As a result of improvement in processing technology, it is now possible to put millions of transistors in an integrated circuit. This increases the amount of processing power of the integrated circuit. However, the processing power may be wasted if there are insufficient input-output (I/O) resources to transfer data to and from the integrated circuit. High speed I/O transceivers alleviate this problem by increasing the data transfer speed of the I/O resources. In order to improve performance, circuit designers have started to integrate high speed transceivers into integrated circuits.

[0003] Another effect of the increase in the number of transistors is that it is possible to place more circuits in an integrated circuit. This means that the complexity and costs of designing an integrated circuit also increase. As a result, many vendors want to be able to sell the same integrated circuit to different markets. Further, users also like to use the same integrated circuit to support different applications because of the high costs of learning and using a complicated integrated circuit. For example, it is desirable for the same product to support a variety of communication protocols, such as Gigabit Ethernet, XAUI, InfiniBand, Fibre Channel, etc. Consequently, there is a need to have a flexible high speed transceiver that can be used for different purposes.

SUMMARY OF THE INVENTION

[0004] The present invention is an integrated circuit that contains a programmable fabric and a plurality of configurable transceivers located at the peripheral of the programmable fabric. The integrated circuit may contain one or more processor cores. The processor core and the transceivers can be connected by a plurality of signal paths that pass through the programmable fabric.

[0005] The integrated circuit contains a plurality of configuration memory cells. Some of the cells are associated with the programmable fabric while the others are associated with the configurable transceivers. By turning these cells on or off, users can select the features that are useful to their products.

[0006] The above summary of the invention is not intended to describe each disclosed embodiment of the present invention. The figures and detailed description that follow provide additional exemplary embodiments and aspects of the present invention.

BRIEF DESCRIPTION OF THE DRAWINGS

[0007] Fig. 1 is a schematic diagram showing an integrated circuit of the present invention.

[0008] Fig. 2 is a schematic diagram of a system of the present invention.

[0009] Fig. 3 is a block diagram showing a transceiver design of the present invention.

DETAILED DESCRIPTION OF THE INVENTION

[0010] A schematic diagram showing an integrated circuit 100 of the present invention is shown in Fig. 1. It contains a plurality of transceivers, such as transceiver 102-104, positioned outside of programmable fabric 106. In one embodiment, programmable fabric 106 comprises field programmable gate array (FPGA) fabric. One or more processor cores, such as a processor core 110, can be optionally

embedded inside programmable fabric 106. An interface layer 112 is used to facilitate communication between embedded processor core 110 and fabric 106. An example of an interface layer is disclosed in a copending patent application entitled "Programmable Gate Array Having Interconnecting Logic To Support Embedded Fixed Logic Circuitry" (serial number 09/968,446 and filed September 28, 2001). Processor core 110 can be connected to one or more transceivers. In Fig. 1, a pair of routed paths 114-115 are used to schematically show the connection from transceiver 102 to processor core 110 through interface layer 112. Processor core 110 is preferably designed to process data in high speed. Thus, transceivers 102-104 are preferably high speed transceivers.

[0011] Processor core 110 can be one of a variety of signal processing devices, such as a microprocessor, network processor, video processor. Note that more than one processor core can be embedded inside programmable fabric 106.

[0012] A user may configure a plurality of circuits in programmable fabric 106. Some of these circuits may communicate with transceivers 102-104. For example, Fig. 1 shows a circuit 118 in programmable fabric106 communicating with transceiver 103.

[0013] In the present invention, at least one of the transceivers (in addition to programmable fabric 106) is configurable. Fig. 2 shows a schematic diagram of a system 200 of the present invention showing the configuration of a programmable logic device. System 200 contains a nonvolatile memory (such as a programmable read-only memory 202) that can be used to deliver configuration bitstreams to FPGA 204. FPGA 204 comprises a plurality of configuration memory cells, shown as reference numeral 210, that are connected to a frame register 206 through a plurality of buses, shown as reference numeral 212. These buses allow frame register 206 to set the states of memory cells 210, thereby configuring FPGA 204. Some of the configuration memory cells are used to configure programmable fabric 106, while other configuration memory

cells are used to configure the transceivers. For example, configuration memory cells in blocks 216 and 218 are used to configure two separate transceivers, such as transceivers 103 and 104 in Fig. 1. Note that there are typically many configuration memory cells associated with a block.

[0014] One difference between programmable fabric 106 and the transceivers 102-104 is that the programmable fabric is intended for implementation of arbitrary logic functions by users (thus the programmable fabric needs to provide almost arbitrary interconnectivity), while the transceiver implements specific functions (high speed serial I/O and associated operations), thus the associated memory cells modify aspects of the operations but not changing the underlying purpose of the circuit. In the programmable fabric, memory cells define logic block functions and interconnections in any desired way to implement any desired logic design. In the transceiver, some memory cells modify the behavior of a given subcomponent, but they do not change the underlying function of that subcomponent. Other memory cells associated with the transceiver modify routing by including or excluding a subcomponent in the transceiver's data path, but they do not alter the general flow of data or the purpose of the data path, that is, to transfer parallel digital data inside the integrated circuit to and from the serial data lines outside the integrated circuit. The advantage of this design is that many of the functions in the transceiver are implemented far more efficiently (higher operating speed, less area and power) than would be possible if they were implemented in the programmable fabric, yet they retain enough flexibility to be of use for a broad range of high speed I/O applications. Some portions of the transceiver are highly specialized and could not be implemented using the programmable fabric or lower speed I/O resources.

[0015] In one embodiment, a portion of the transceiver can be configured using the configuration memory cells and

another portion can be driven by configured logic in the programmable fabric. It is, of course, possible that the transceiver is controlled entirely by the configuration memory cells.

[0016] Fig. 3 is a block diagram showing one transceiver 130 that can be fabricated in integrated circuit 100 of Fig. 1. Transceiver 130 interfaces with an external device (not shown) through an output port 132 that supports differential output signals and an input port 134 that receives differential input signals. Transceiver 130 interfaces with programmable fabric 106 through a transmit data path 136, a receive data path 138, a plurality of clock signals (shown collectively as a signal line 140), a CRC (cyclic redundancy code) status signal 142, and a loss of synchronization signal 144. In one embodiment, the width of the data paths 136 and 138 can be independently configurably selected to be 1, 2 or 4 bytes. This allows engineering trade-offs in programmable fabric 106 between a wide data path with a low clock frequency versus a narrow data path with a high clock frequency. More detailed description of the configurable datapath can be found in a copending patent application entitled "Variable Data Width Operation In Multi-Gigabit Transceivers On A Programmable Logic Device," filed concurrently. This patent application is incorporated herein by reference.

[0017] Note that there may be other signals and subcomponents in a transceiver. Only the most important signals and subcomponents are shown in Fig. 3.

[0018] The transmitter side 150 of transceiver 130 is first described. Digital data on data path 136 is processed by a bypassable CRC generator 152. This generator computes and inserts a commonly used 32-bit CRC into the data packets being transmitted. Different protocols handle data in different ways, and CRC generator 152 needs to recognize data packet boundaries so that it can perform CRC computation on

appropriate set of data. CRC generator 152 may have the following configuration options:

(a) use or bypass this CRC generator;

(b) a value to use to corrupt the CRC computation in order to test detection of CRC errors in a remote receiver;

(c) a choice of transmission standard to support, e.g., Fibre Channel, Gigabit Ethernet, InfiniBand, or a user defined standard; and

(d) for the user defined standard, the start-of-packet and end-of-packet control characters.

[0019] More detailed description of CRC generator 152 can be found in a copending patent application entitled "Network Physical Layer with Embedded Multi-Standard CRC Generator," filed concurrently. This patent application is incorporated herein by reference.

[0020] The resulted data is delivered to a bypassble encoder 154. In one embodiment, encoder 154 is an 8B/10B encoder. It uses the same 256 data characters and 12 control characters that are used for Gigabit Ethernet, XAUI, Fibre Channel, and InfiniBand. It accepts 8 bits of data along with a K-character signal for a total of 9 bits per character applied. If the K-character signal is "High", the data will be encoded into one of the 12 possible K-characters available in the 8B/10B code. If the K-character input is "Low", the 8 bits will be encoded as standard data. If the K-character input is "High", and a user applies other than one of the pre-assigned combinations, an error signal can be generated. The 8B/10B encoder may be initialized with a user-configured running disparity.

[0021] In one embodiment, the programmable fabric may control the following options for encoder 154:

(a) use or bypass this encoder; and

(b) modify the maintenance of the running disparity (this can be used to generate data streams with unusual, e.g., intentionally erroneous, running disparity).

[0022] The encoded data is delivered to a transmit FIFO buffer 156. This buffer provides a smooth interface between encoder 154, which is controlled by a clock signal generated by the programmable fabric 106 on line 140, and a serializer 158, which is controlled by a reference clock generated by a transmit clock generator 160. These two clock signals are frequency locked, but may not have the same phase. In this embodiment, the reference clock has superior jitter characteristics while the clock signal on line 140 has better clock skew characteristics with respect to other clock signals in programmable fabric 106. Transmit FIFO buffer 156 absorbs phase differences between these two frequency-locked clock signals. In one embodiment, FIFO buffer 156 has a depth of four, and it can detect overflow and under-flow conditions. FIFO buffer 156 may be configured to be used or bypassed in transceiver 130.

[0023] The data in FIFO buffer 156 is delivered to serializer 158, which multiplexes parallel digital data to a serial bit stream for transmission over a serial link. The serial bit stream is sent to a transmit buffer 162 that drives the serial bit stream onto a pair of differential serial output connections in output port 132. A configuration option of serializer 158 is to transmit 20 bits (high speed) or 10 bits (low speed) of data per reference clock cycle. More detailed description of this aspect of the invention can be found in a copending patent application entitled "Method And Apparatus For Operating A Transceiver In Different Data Rates," filed concurrently. This patent application is incorporated herein by reference.

[0024] Turning now to the receiver side 170, transceiver 130 contains a receiver clock generator 176 that generates a reference clock signal for receiver side 170. Transceiver 130 contains a receive buffer 172 that accepts serial data from differential input port 134. The data is fed to a clock-data recovery block 173, which uses transitions on the serial differential input (through input port 134) to determine the

frequency and phase of the incoming serial data. This information is used to time the reception of the data. The process is called clock and data recovery. The recovered data is delivered to a deserializer 174, which converts the received serial bit stream into parallel digital data. A configuration option of deserializer 174 is to receive 20 bits (high speed) or 10 bits (low speed) of data per clock cycle.

Deserializer 174 also performs a comma detection function. In some decoding algorithm (such as the 8B/10B encoding), a "comma" is a distinguished pattern that is guaranteed to occur only left-justified within a byte. For example, there are two comma patterns in 8B/10B decoding, a "plus" comma and a "minus" comma. Detection of a comma then serves to define the byte alignment within the received serial bit stream. Configuration options for comma detection can be:

(a) alternative comma definitions;

(b) option to raise asynchronous "comma detect" flag on plus comma only, minus comma only, both, or neither; and

(c) option to force comma alignment on half-word boundary.

(d) In one embodiment, programmable fabric 106 may send a signal to control whether the comma detection circuit realigns the byte boundary on recognizing plus comma, minus comma, both, or neither.

[0025] The parallel data is then passed to a decoder 178. Decoder 178 decodes digital data that has previously been encoded by a corresponding encoder. Decoder 178 may have the following configuration options:

(a) use or bypass this decoder; and

(b) option to raise synchronous "comma" flag (status bit attached to each received byte at the transceiver-programmable fabric interface) on plus comma only, minus comma only, both, or neither. Also option to set this flag for valid commas only (several invalid comma patterns are also possible).

[0026] The decoded data is then sent to an elastic buffer 182, which carries out channel bonding and clock correction operations in conjunction with a channel bonding and clock correction controller 184. Elastic buffer 182 may have the following configuration options:

(a) use or bypass the elastic buffer;

(b) use or inhibit clock correction;

(c) threshold value for flagging buffer overflow or underflow; and

(d) options related to the operation of the elastic buffer (clock correction and channel bonding), such as the choice of channel bonding modes, the selection of a selectable number of channel bonding sequences of a selectable length matching a selectable byte value (8-bit or 10 bit), and similar selection for clock correction sequences.

[0027] The data in elastic buffer 182 is delivered to the programmable fabric through data path 138. The data width can be configurably selected to be 1, 2 or 4 bytes.

[0028] Elastic buffer 182 can be optionally connected to a CRC verification block 186. This block verifies the commonly used 32-bit CRC that is expected to appear at the end of received data packets. A signal may be delivered to the programmable fabric on line 142 indicating the result of the verification. In order to perform CRC, verification block 186 should recognize data packet boundaries (similar to CRC generation block 152 described above). CRC Verification block 186 may have the following configuration options:

(a) use or bypass this CRC verification block;

(b) a choice of transmission standard to support, e.g., Fibre Channel, Gigabit Ethernet, InfiniBand, or a user defined standard; and

(c) for the user defined standard, the start-of-packet and end-of-packet control characters.

[0029] Transceiver 130 can optionally include a loss of synchronization detector 188. It interprets outputs of comma detection 174, decoder 178, and elastic buffer 182 to make a

determination of whether the incoming byte stream is in sync. A signal can be delivered to the programmable fabric on line 144. Loss of synchronization detector 188 may have the following configuration options:

(a) use or bypass this detector;

(b) option to set number of invalid characters that cause "loss of sync" status; and

(c) option to set the number of valid characters that negates effect of one invalid character for determination of loss of synchronization.

[0030] In the event that loss of synchronization detector 188 is bypassed, status information of decoder 178 and elastic buffer 182 may need to be delivered to the programmable fabric using other status lines.

[0031] Referring back to Fig. 2, the configuration information on the configurable transceivers of the present invention can be stored in PROM 202. A user can select different options by delivering different configuration bitstreams to FPGA 204.

[0032] Those having skill in the relevant arts of the invention will now perceive various modifications and additions which may be made as a result of the disclosure herein. Accordingly, all such modifications and additions are deemed to be within the scope of the invention, which is to be limited only by the appended claims and their equivalents.